SOFTWARE COMMUNICATIONS ARCHITECTURE SPECIFICATION

APPENDIX A: GLOSSARY

01 October 2012
Version: 4.0.1

Prepared by:

Joint Tactical Networking Center (JTNC)
33000 Nixie Way
San Diego, CA 92147-5110

REVISION HISTORY

Version	Revision	Date
Next <Draft>	Initial Draft Release	30 November 2010
Next <Draft> 1.0.0.2	Applied SCA Next Errata Sheet v2.0	14 September 2011
Candidate Release	Initial Release	27 December 2011
4.0	ICWG Approved Release	28 February 2012
4.0.1	Incorporated transition to JTNC and applied SCA 4.0 Errata Sheet v1.0	01 October 2012

APPENDIX A GLOSSARY

A.1 ABBREVIATIONS AND ACRONYMS

Abbreviation	Definition
ADD	Application Deployment Descriptor
AEP	Application Environment Profile
AIX	IBM's Unix Implementation
API	Application Program Interface
BIT	Built-In Tests
BSDi	Berkeley Software Design Inc.
CA	Certification Authority
CF	Core Framework
CFCC	CF Control Components
CORBA	Common Object Request Broker Architecture
DCD	Device Configuration Descriptor
DDS	Data Distribution Service
DMD	Domain Manager Configuration Descriptor
DPD	Device Package Descriptor
DSP	Digital Signal Processor
DTD	Document Type Definition
FIFO	First In First Out
FPGA	Field Programmable Gate Array
GPP	General Purpose Processor
HCI	Human-Computer Interface
HPUX	Hewlett-Packard UniX
I/O	Input / Output
ICWG	Interface Control Working Group
ID	Identification, Identifier
IDL	Interface Definition Language
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IIOP	Internet Inter-ORB Protocol
IOR	Interoperable Object Reference
IP	Internet Protocol
IRIX	Silicon Graphics International Corp. UNIX-like Operating System
ISO	International Organization for Standardization

Abbreviation	Definition
J2EE	Java 2 Enterprise Edition
JTNC	Joint Tactical Networking Center
JTRS	Joint Tactical Radio System
LW	Light Weight
LwAEP	Lightweight AEP
MAN	Mandatory
MHAL	Modem Hardware Abstraction Layer
MVS	Multiple Virtual Storage
N/A	Not Applicable
NRQ	Not Required
ODT	Open Document Text
OE	Operating Environment
OMG	Object Management Group
ORB	Object Request Broker
OS	Operating System
PDD	Platform Deployment Descriptor
PIM	Platform Independent Model
POA	Portable Object Adapter
POSIX	Portable Operating System Interface
PRF	Properties Descriptor
PRI	Primary
PRT	Partially Required
PSM	Platform Specific Model
RF	Radio Frequency
RT	Real Time
RTCORBA	Real Time CORBA
RTOS	Real Time Operating System
RTXC	Real-Time Executive in C
SA	Specification Authority
SAD	Software Assembly Descriptor
SCA	Software Communications Architecture
SCD	Software Component Descriptor
SCO	Santa Cruz Operation
SDR	Software Defined Radio
SPD	Software Package Descriptor
SW	Software
TA	Test and Evaluation Authority

Abbreviation	Definition
ULW	Ultra Light Weight
UOF	Units of Functionality
UML	Unified Modeling Language
UTC	Coordinated Universal Time
UUID	Universal Unique Identifier
VM	Virtual Machine
VMS	Virtual Memory System
W3C	World Wide Web Consortium
XML	eXtensible Markup Language
XSD	XML Schema Definition

A.2 DEFINITIONS

application

An application consists of one or more software modules which implement the Base Application Interfaces and are identified within a Software Assembly Descriptor (SAD) file. When loaded and executed, these modules create one or more components which comprise the application.

application factory

An implementation of the *ApplicationFactory* interface, an application factory is used to create an instance of an application. The domain manager creates an application factory for each SAD that is installed.

application manager

An implementation of the *Application* interface, an application manager, created by the ApplicationFactory, provides access to a specific application instantiated on the system.

assemblycontroller

The *assemblycontroller* element of the SAD indicates the component that is the main resource controller for an application.

Application Deployment Descriptor (ADD)

An ADD is an element of the Domain Profile that defines the channel deployment precedence order for the application.

Child Device

A device intended to be strongly associated to a related parent device. See Parent Device.

Component

A component is an autonomous unit whose manifestation is replaceable within its environment. A component exposes a set of ports that define the component specification in terms of provided and required interfaces. As such, a component serves as a type, whose conformance is defined by these provided and required interfaces (encompassing both their static as well as dynamic semantics). An SCA component identifies its interfaces within a Software Component Descriptor.

Core Framework (CF)

The CF is the set of open application-layer interfaces and services defined within this specification. The CF provides the essential ("core") set of open software interfaces and profiles that provide for the deployment, management, interconnection, and intercommunication of software application components in an embedded, distributed-computing communication system. In this sense, all SCA interfaces are part of the CF.

Destroy

The act of releasing / terminating a software object or component.

device

An implementation of the *Device* interface, a device is an abstraction of a hardware device that defines the capabilities, attributes, and interfaces for that device.

Hardware device refers to a physical hardware element (typically a module performing a function or set of functions).

Device Configuration Descriptor (DCD)

The DCD is an element of the Domain Profile that identifies all devices and services associated with a device manager, by referencing its associated Software Package Descriptors (SPDs). The DCD also defines properties of the specific device manager, enumerates the needed connections to services (e.g. file systems), and provides additional information on how to locate the domain manager

device manager

An implementation of the *DeviceManager* interface, a device manager contains complete knowledge of a set of devices and/or services and registers with the domain manager to assure that the domain manager has complete cognizance of the system. A device manager may have an associated file system (or file manager to support multiple file systems).

Device Package Descriptor (DPD)

A DPD is an element of the Domain Profile that contains information about a hardware device. The DPD has properties that define specific information (manufacturer, model number, serial number, etc.) about the device.

Device Profile

The Device Profile is the set of XML files within the Domain Profile which describes a hardware device. The Device Profile contains a DPD, a DCD, and an optional Properties Descriptor (PRF). Information about the software associated with this hardware device is found in the associated Software Profile.

Domain

A Domain defines a set of hardware devices and available applications under the control of a single domain manager component.

Domain Manager Configuration Descriptor (DMD)

A DMD is an element of the Domain Profile that provides the location of the SPD file for a specific domain manager. The DMD also specifies connections to other software components (e.g. services) which are required by the domain manager. The DMD may also reference a Platform Deployment Descriptor (PDD) that describes the channels for a platform.

domain manager

An implementation of the *DomainManager* interface, a domain manager manages the complete set of available hardware devices and applications. It is responsible for the set-up and shut-down of applications and for allocating resources, devices, and other components to hardware devices.

Domain Profile

The hardware devices and software components that make up an SCA system domain are described by a set of XML files that are collectively referred to as a Domain Profile. The domain manager uses the Domain Profile to build its internal information base from the descriptions of the individual hardware devices, software components, and application assemblies under its control.

File and Filename

The terms "file" and "filename" as used in the SCA, refer to both a "plain file" (equivalent to a POSIX "regular file") and a directory. An explicit reference is made within the text when referring to only one of these.

Incoming Domain Management Event Channel

Incoming Domain Management Event Channel is an event channel that is internal to the domain and is used by domain’s components to send events to the domain management components (ApplicationManagerComponent, ApplicationFactoryComponent, and DomainManagerComponent).

Non-SCA Component

A non-SCA Component is a component that does not realize any of the SCA defined interfaces.

Non-SCA Service

Non-SCA services are those other than log, file system, and Event Services launched by a Device Manager and can be managed by the framework through the CF based interfaces. Non-SCA service definitions consist of APIs, behavior, state, priority and additional information that provide the contract between the Service Provider and the Service User.

Obtainable Port

An obtainable port that has its lifecycle tied to the lifecycle of the connection. An obtainable port is not registered with the component; therefore the CF must retrieve the port when making the connection. Obtainable ports are explicitly released by CF when the connection is torn down.

Operating Environment (OE)

The OE provides the capabilities to host waveforms and allow them to access system resources (i.e. manage and execute SCA components) and consists of the Operating System, Transfer Mechanism, CF Control and Platform Devices and Services.

Outgoing Domain Management Event Channel

Outgoing Domain Management Event Channel is an event channel that is external to the domain and is used by external domain's components (e.g., HCI) to receive events by domain management components, ApplicationManagerComponent. ApplicationFactoryComponent and DomainManagerComponent.

Parent Device

A parent device uses the *AggregateDevice* interface and is composed of one or more child devices. The parent device and its children are strongly associated and have the same lifetime (i.e. removal of the parent device removes all child devices).

Pathname

An "absolute pathname" is a pathname which starts with a "/" (forward slash) character - a "relative pathname" does not have the leading "/" character. A "path prefix" is a pathname which refers to a directory and thus does not include the name of a plain file. Pathnames are used in accordance with the POSIX specification definition and may reference either a plain file or a directory.

Platform Independent Model (PIM)

A model of a subsystem that contains no information specific to a specific platform, or the underlying technology used to realize it.

Port

An implementation of the *Port* interface which identifies a source (Provides Port) or a sink (Uses Port) for data and/or commands.

Profile Descriptor

A Profile Descriptor is an element of the Domain Profile that contains an absolute pathname for either a SPD, SAD, DMD, or a DCD.

Platform Deployment Descriptor (PDD)

A PDD is an element of the Domain Profile that describes the channels for a platform.

Properties Descriptor (PRF)

A PRF is an element of the Domain Profile that contains the properties applicable to a software package or a device package such as configuration, test, execute, and allocation types.

Property

An SCA Property is a variable that contains a value of a specific type. Configuration Properties are parameters to the *configure* and *query* operations of the *PropertySet* interface. Allocation Properties define the capabilities required of a device by a resource.

Platform Specific Model (PSM)

A subsystem model that incorporates technology specific information which is used in its realization on a specific platform.

Registered Port

A port that has its lifecycle tied to the lifecycle of the component. A registered port is registered (i.e. associated) with the component in an implementation specific manner as part of component instantiation or initialization, thus the CF does not attempt to retrieve the port when making a connection. Registered provides ports are not explicitly released by CF except as part of *releaseObject* operation behavior

Release (from the Operating Environment)

When an object is released, it is no longer able to process object requests; e.g. if using CORBA then its object reference unavailable to other objects. In the SCA, a component is removed from the OE and OE resources consumed by a component are returned back to the OE. After a component is removed from the OE, a client is unable to communicate with the component.

resource

An implementation of the *Resource* interface, a resource provides a common API for the control and configuration of a software component.

Software Assembly Descriptor (SAD)

A SAD is an element of the Domain Profile that contains information about the components that make up an application.

Software Component Descriptor (SCD)

A SCD is an element of the Domain Profile that contains information about a specific SCA software component (e.g. ResourceComponent, ComponentFactoryComponent, or ComponentBaseDevice).

Software Package Descriptor (SPD)

A SPD is an element of the Domain Profile that identifies a software component implementation(s). General information about a software package, such as the name, author, property file, and implementation code information and hardware and/or software dependencies are contained in a SPD file.

Software Profile

A Software Profile is a set of Domain Profile files which pertain to a specific SCA component. All software profiles for components include a SPD and a SCD (as well as optional PRF files), but the other files contained in the profile depend on the SCA component in question. The profile for an application will include a SAD, a domain manager profile includes a DMD, and a profile for a device manager will include a DCD.

Transfer Mechanism

The transfer mechanism structure may be comprised of object request semantics, transfer and message syntax, and transports. The SCA leverages transfer mechanisms to provide standardized client/server operations. Client/server communications may be co-located or distributed across different processors.

Waveform

A waveform is the set of transformations applied to information that is transmitted over the air and the corresponding set of transformations to convert received signals back to their information content.